



Support for Emerging Climate and Forecast Grid Conventions in the Earth System Modeling Framework

Tony Wong

University of Colorado at Boulder

Robert Oehmke

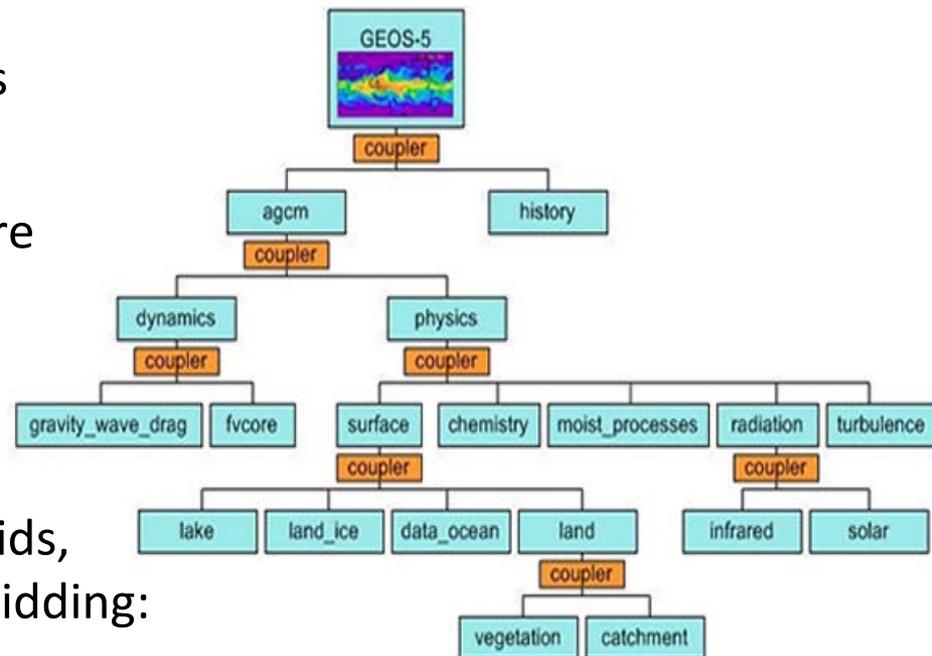
NOAA/CIRES



Outline

- ESMF overview
- Grid conventions and you!
- Gridspec
- New ESMF features: Gridspec
- UGrid
- New ESMF features: UGrid
- Exciting conclusion!

- Breaks model into components
- Provides standard infrastructure
- Provides many utilities
- Because we're working with grids, the most relevant utility is regriding:
 - Integrated
 - File-based off-line
 - ESMF_RegridWeightGen



Applications of information layer

- Parallel generation and application of interpolation weights
- Run-time compliance checking of metadata and time behavior
- Fast parallel I/O
- Redistribution and other parallel communications
- Automated documentation of models and simulations (new)
- Ability to run components in workflows and as web services (new)

Structured model information stored in ESMF wrappers

ESMF data structures

Standard metadata

Attributes: CF conventions, ISO standards, METAFOR Common Information Model

Standard data structures

Component

Field

Grid

Clock

User data is referenced or copied into ESMF structures

Native model data structures

modules

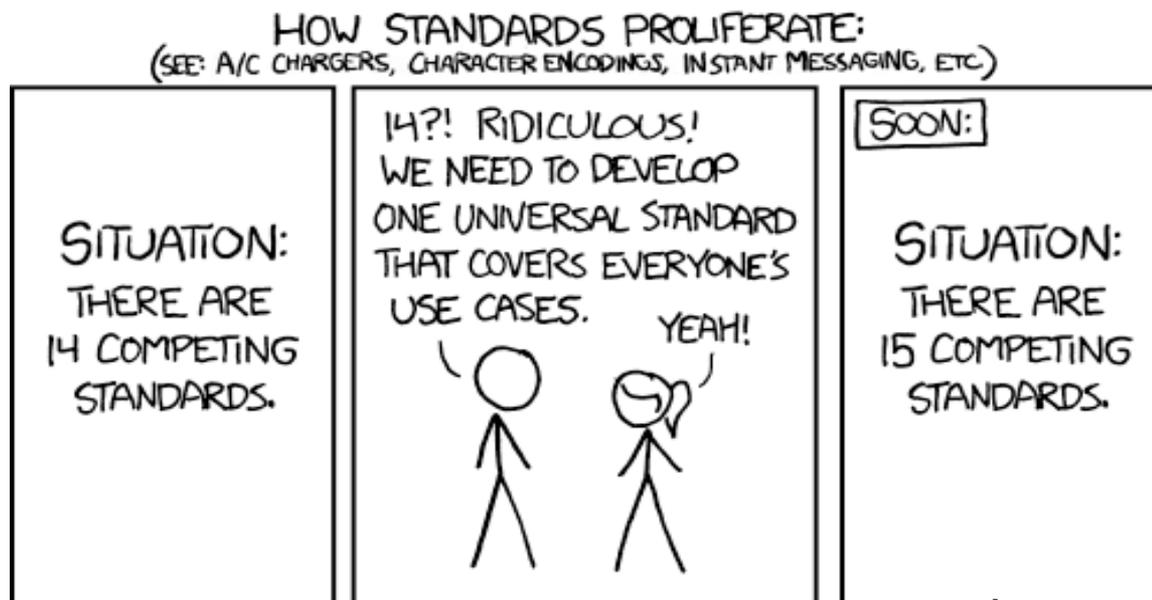
fields

grids

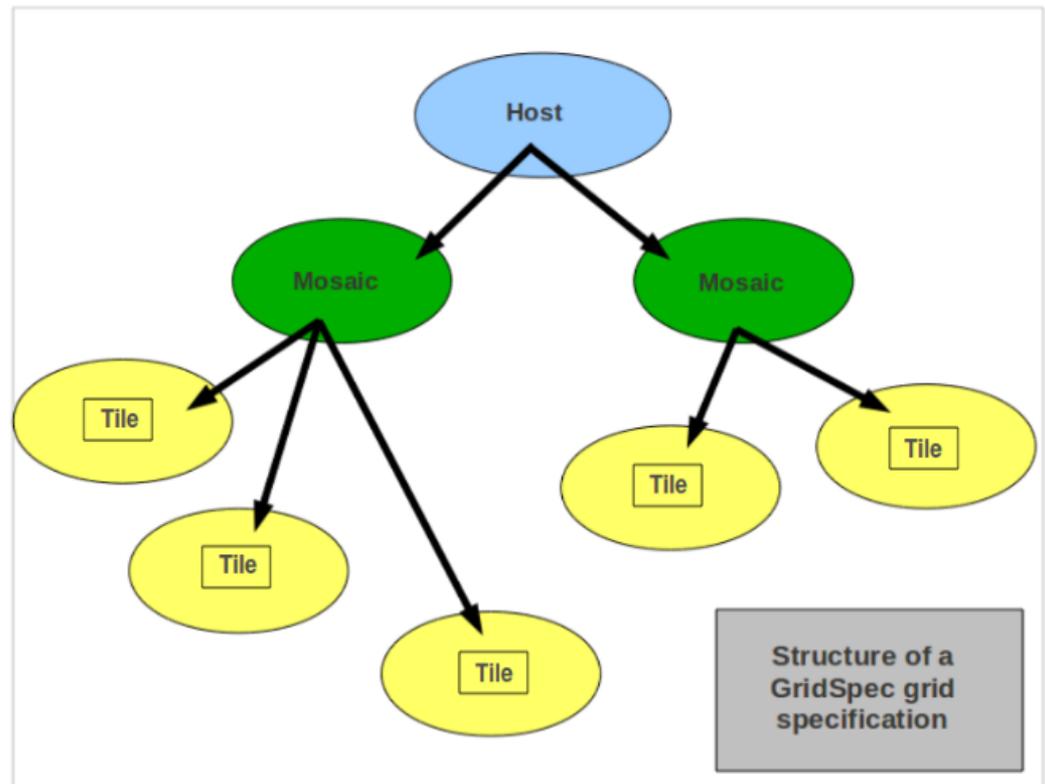
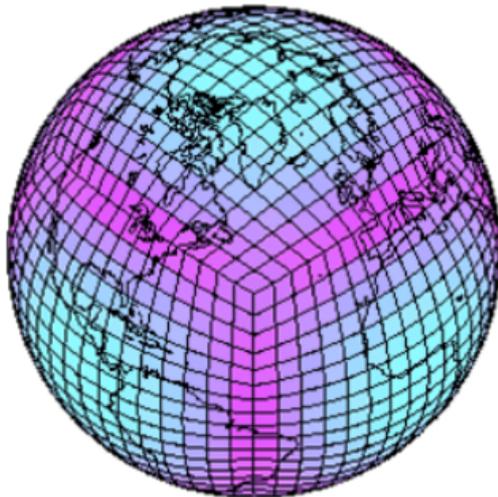
timekeeping

- Encourages interaction and collaboration between agencies, research centers and universities:
 - Easier to collaborate and exchange model components
 - Easier to visualize and compare results
 - Easier to save time and efforts by reusing code

- Allows easier sharing, analyzing, remapping, visualization and inter-comparing of data and results between modeling efforts
- Provides easy access to metadata describing the grids and methods used
- Communal efforts



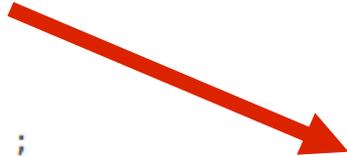
- Proposed climate and forecasting (CF) convention for grids composed of a set of logically rectangular tiles [1]





Gridspec

```
netcdf GridspecTest2D_host {
dimensions:
  string = 256 ;
  ngrid = 1 ;
variables:
  char mosaic_filename(string) ;
  char tile_names(ngrid, string) ;
  char tile_filenames(ngrid, string) ;
// global attributes:
  :gridspec_file_type = "gridspec_host_file" ;
}
```



```
netcdf GridspecTest2D_mosaic {
dimensions:
  string = 256 ;
  ndims = 2 ;
  ngrid = 1 ;
variables:
  char gridspec_coordinate_names(ndims, string) ;
  char tile_names(ngrid, string) ;
// global attributes:
  :gridspec_file_type = "gridspec_mosaic_file" ;
}
```

```
netcdf GridspecTest2D_tile {
dimensions:
  string = 256 ;
  dim1 = 10 ;
  dim2 = 10 ;
variables:
  double coord1(dim1, dim2) ;
  coord1:units = "degrees_coord1" ;
  double coord2(dim1, dim2) ;
  coord2:units = "degrees_coord2" ;
// global attributes:
  :gridspec_file_type = "gridspec_tile_file" ;
  :gridspec_tile_name = "GridspecTest2D_tile" ;
}
```





New ESMF support for Gridspec

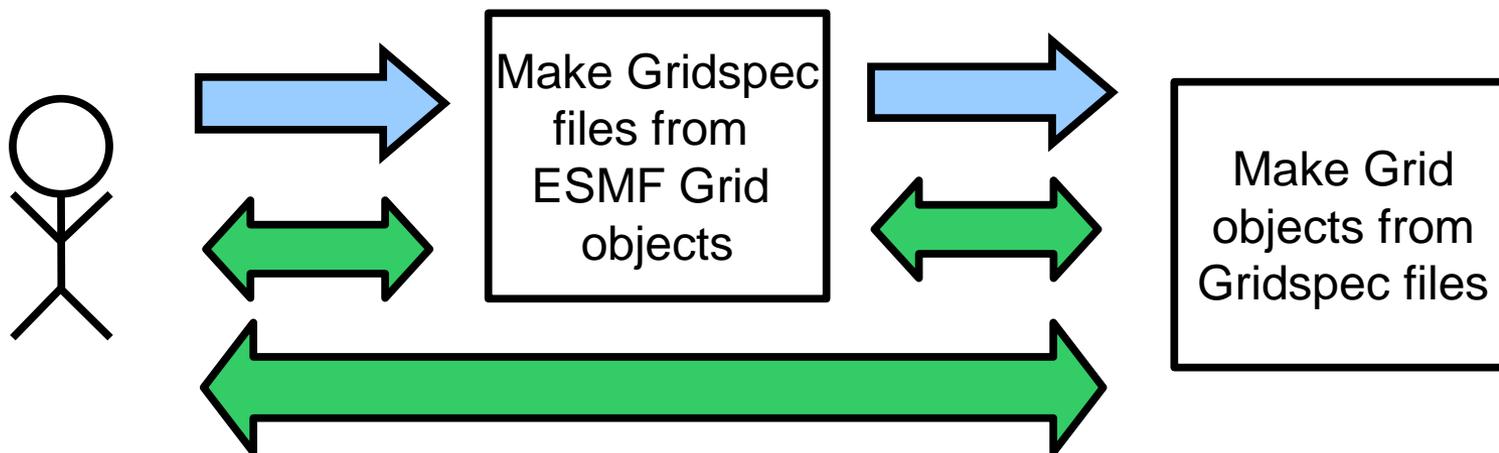
- 1D-4D
- Corner or center staggering
- Write user-specified (with all above options) single-patch tile-mosaic-host file trios

```
call ESMF_GridWrite("GridspecUser", 2, (/3,3/), coords=coord2D, &  
                  staggerLoc=localStaggerLoc, gridOut=grid, rc=localrc)
```

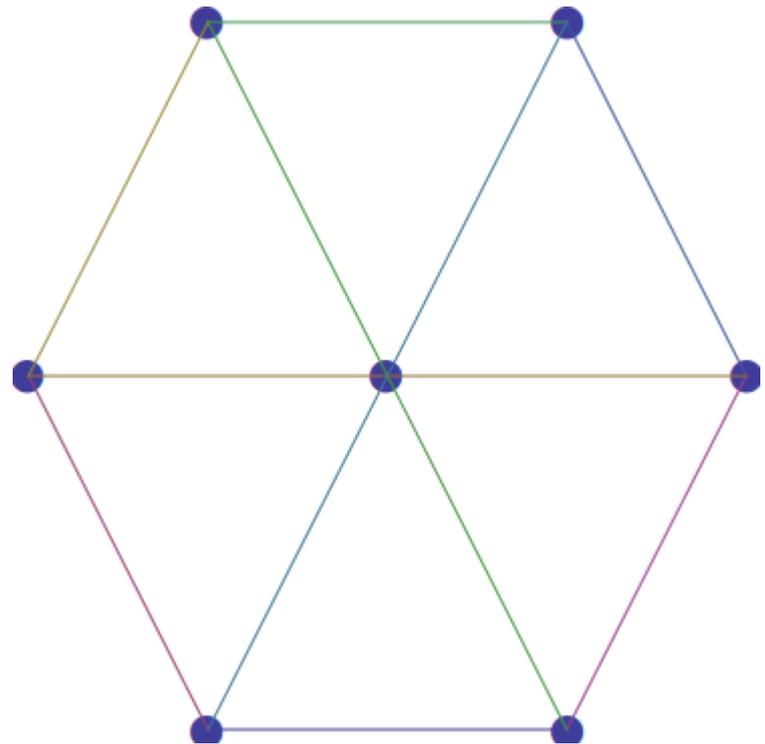
- Read single-patch Gridspec format netCDF grid files and create ESMF Grid objects

```
grid = ESMF_GridCreate(grid_filename="GridspecTest2D_grid.nc", &  
                      mosaic_filename="GridspecTest2D_mosaic.nc", &  
                      regDecomp=(/1,1/), addCornerStagger=.true., rc=localrc)
```

- Gridspec file generation:
 - Create 1-4D single-patch tile-mosaic-host trios for both stagger locations
 - Check coordinate information
 - If that runs smoothly, then...
- ESMF Grid generation from Gridspec files:
 - Create ESMF Grid objects from the files written in the previous step
 - Check coordinate information



- Proposed CF convention for unstructured grids [2]
- All coordinate and connectivity information stored in one netCDF file



Example unstructured grid



UGrid

```
netcdf esmf_ugridtest2d {
dimensions:
    nNodes = 8 ;
    nEdges = 12 ;
    nFaces = 5 ;
    nFaceLinks = 4 ;
    nMax_FaceNodes = 4 ;
    Two = 2 ;
variables:
    int esmf_ugridtest2d ;
    esmf_ugridtest2d:standard_name = "mesh_topology" ;
    esmf_ugridtest2d:long_name = "Topology data of 2D unstructured mesh" ;
    esmf_ugridtest2d:dimension = 2 ;
    esmf_ugridtest2d:locations = "node edge face" ;
    esmf_ugridtest2d:node_coordinates = "node_x node_y" ;
    esmf_ugridtest2d:edge_coordinates = "edge_x edge_y" ;
    esmf_ugridtest2d:face_coordinates = "face_x face_y" ;
    esmf_ugridtest2d:edge_node_connectivity = "edge_nodes" ;
    esmf_ugridtest2d:face_node_connectivity = "face_nodes" ;
    esmf_ugridtest2d:face_edge_connectivity = "face_edges" ;
    esmf_ugridtest2d:face_face_connectivity = "face_links" ;
    double node_x(nNodes) ;
    node_x:standard_name = "longitude" ;
    node_x:long_name = "Longitude of 2D mesh nodes" ;
    node_x:units = "degrees_east" ;
    double node_y(nNodes) ;
    node_y:standard_name = "latitude" ;
    node_y:long_name = "Latitude of 2D mesh nodes" ;
    node_y:units = "degrees_north" ;
    double edge_x(nEdges) ;
    edge_x:standard_name = "longitude" ;
    edge_x:long_name = "Characteristic longitude of 2D mesh edges (midpoint)" ;
    edge_x:units = "degrees_east" ;
    double edge_y(nEdges) ;
```

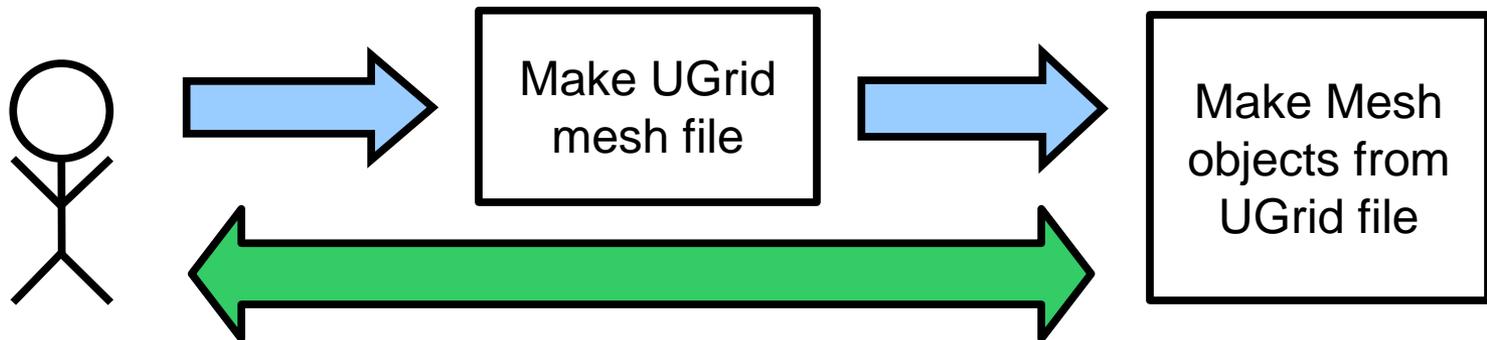


New ESMF support for UGrid

- 2D and 3D
- Read UGrid file, create ESMF Mesh object

```
mesh = ESMF_MeshCreate(filename="MeshExample.nc", &  
                        filetype=ESMF_FILEFORMAT_UGRID, rc=rclocal)
```

- Created 2D UGrid-conformant test mesh
- Use ESMF Mesh utilities to check mesh properties match
 - Connectivity
 - Coordinate information





Now possible...

- Easily create ESMF Grid object from a Gridspec format file
 - Users may access ESMF utilities such as regridding
- Easily write ESMF Grid object to a Gridspec-conformant file, including mosaic and host files
 - Users of the ESMF may easily output data or grid definitions into this convention
- Easily create ESMF Mesh object from a UGrid format file
 - Users may access ESMF utilities such as regridding
- Allows users of the ESMF to work with these upcoming grid conventions, encouraging collaboration by making it easy to share information, analyze, inter-compare and visualize results



Future work

- Extend Gridspec support to multi-patch mosaics and multi-mosaic hosts
- Extend Gridspec support to higher dimensions
- Add Gridspec and UGrid formats as options to ESMF_RegridWeightGen application

- [1] Balaji, V. “Gridspec: A standard for the description of grids in Earth System models,” 2007.
- [2] Jagers, Bert. “Deltares CF proposal for Unstructured Grid data model,” 2011.
- [3] Hill, C., C. DeLuca, V. Balaji, M. Suarez, and A. da Silva (2004). Architecture of the Earth System Modeling Framework. Computing in Science and Engineering, Volume 6, Number 1, pp. 18-28.